Atividade

Siga as instruções

2021002252@ifam.edu.br Alternar conta

Rascunho salvo.

Seu e-mail será registrado quando você enviar este formulário.

*Obrigatório

Defina Árvore Binária com as suas respectivas propriedades. *

Uma Árvore Binária é uma estrutura de dados composta por:
1 - Nó Raiz (nó unico que irá direcionar os demais elementos para sub-arvores);
2 - Nós (elementos inseridos depois do Nó Raiz e que podem possuir até 2 Nós Folhas);
3 - Nós Folhas (elementos que encontram-se no último nível da árvore, e cujos ponteiros apontam para NULL).

Dentre suas características, destacam-se:
1 - Em cada Nó, o menor valor é direcionado a esquerda, e o maior valor é direcionado a direita;
2 - Uma árvore pode se sub-dividir em 2 sub-arvores: Sub-Arvore Esquerda (cujos valores são menores que o Nó Raiz) e Sub-Arvore Direita (cujos valores são maiores que o Nó Raiz).
3 - A altura da árvore dá-se da seguinte maneira: A altura começa no 0, e vai aumentando conforme descemos a árvore binária, até os Nós Folhas.

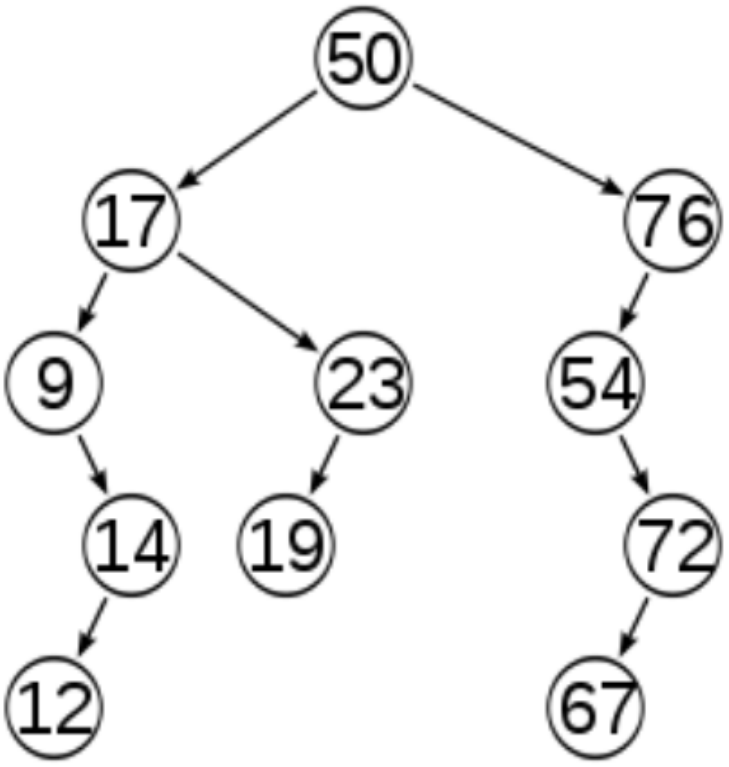
Explique a estrutura de dados NO de uma Árvore Binária. *

Tal estrutura possui 3 elementos básicos, um tipo primitivo de dados (inteiro, caracter, float - é onde guardamos a informação em nosso Nó), e 2 ponteiros (1 para o Nó esquerdo e 1 para o Nó direito).

Defina Função/Procedimento Recursivo. *

Uma Função/Procedimento Recursiva é uma função/procedimento que, durante a sua execução, ela irá se auto chamar. É como um loop infinito (se não houver uma condição para encerra-la, sua operação nunca acabará).

Figura 01



Com base na Figura 01. Quais são os NO(s) raízes da árvore binária, da sub-árvore esquerda e da sub-árvore direita? *

50; 17 e 76

Com base na Figura 01. Qual é o nível do NO 19? *

3

Com base na Figura 01. Qual é a altura da árvore binária? *

4

Com base na Figura 01. Identifique os NO(s) folhas. *

12, 19 e 67

Com base na Figura 01. Exiba os elementos da Árvore Binária na sequência pré-ordem. *

50, 17, 9, 14, 12, 23, 19, 76, 54, 72, 67

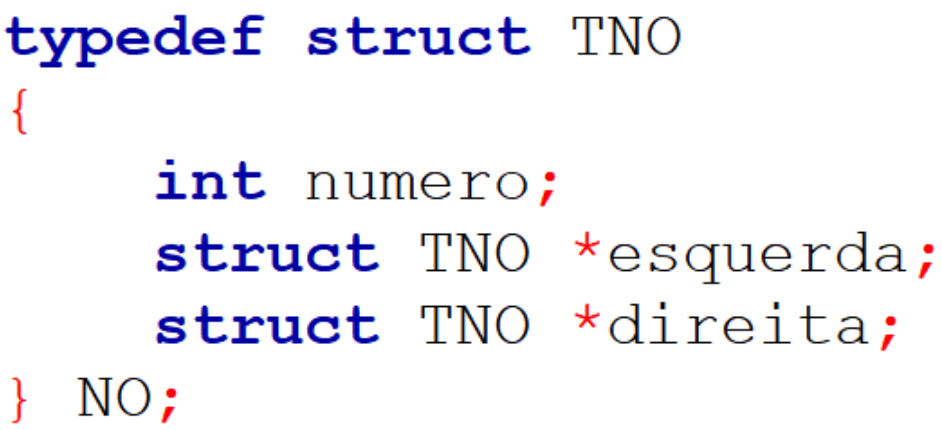
Com base na Figura 01. Exiba os elementos da Árvore Binária na sequência em ordem. *

9, 12, 14, 17, 19, 23, 50, 54, 67, 72, 76

Com base na Figura 01. Exiba os elementos da Árvore Binária na sequência pós-ordem. *

12, 14, 9, 19, 23, 17, 67, 72, 54, 76, 50;

Figura 02



Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: Criar árvore binária. *

```
void iniciarArvore(NO **raiz) {
  *raiz = NULL;
}
```

Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: inserir. *

```
void inserir(NO **raiz, int p_numero) {
  if (*raiz == NULL) {
     // alocar memoria para a estrutura da arvore
     *raiz = (NO *)malloc(sizeof(NO))
    // campos da estrutura da arvore recebem um valor;
    (*raiz)->esquerda = NULL;
    (*raiz)->direita = NULL;
    (*raiz)->numero = p_numero;
  }
  else {
     if (p_numero < ((*raiz)->numero)) {
        inserir(&((*raiz)->esquerda), p_numero);
     }
     else {
        inserir(&((*raiz)->direita), p_numero);
     }
  }
}
```

Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: imprimir pré-ordem. *

```
void imprimirPreOrdem(NO *raiz) {
  printf(" (%d) ", raiz->numero);
  imprimirPreOrdem(raiz->esquerda);
  imprimirPreOrdem(raiz->direita);
}
```

Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: imprimir em ordem. *

```
void imprimirEmOrdem(NO *raiz) {
  imprimirPreOrdem(raiz->esquerda);
  printf(" (%d) ", raiz->numero);
  imprimirPreOrdem(raiz->direita);
}
```

Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: imprimir pós-ordem. *

```
void imprimirPosOrdem(NO *raiz) {
  imprimirPreOrdem(raiz->esquerda);
  imprimirPreOrdem(raiz->direita);
  printf(" (%d) ", raiz->numero);
}
```

Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: contar NO da árvore binária. *

```
int contarNO(NO *raiz) {
  if (raiz == NULL) {
    return 0;
  }
  else {
    return 1 + contarNO(raiz->esquerda) + contarNO(raiz->direita);
  }
}
```

Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: calcular a altura da árvore binária. *

```
int maiorValor(int v1, int v2) {
  if (v1 > v2) {
    return v1;
  }
  else {
    return v2;
  }
}

int calcularAltura(NO *raiz) {
  if (raiz == NULL) {
    return 0;
  }
  else {
    return 1 + maiorValor(calcularAltura(raiz->esquerda), calcularAltura(raiz->direita));
  }
}
```

Com base na Figura 02. Implemente, utilizando a linguagem de programação C, a rotina: calcular a quantidade de NO folhas da árvore binária. *

```
int contarFolhas(NO *raiz) {
  if (raiz == NULL) {
    return 0;
  }
  if (raiz->esquerda == NULL && raiz->direita == NULL) {
    return 1;
  }
  return contarFolhas(raiz->esquerda) + contarFolhas(raiz->direita);
}
```

Enviar Limpar formulário